# Implementing SBC Redundancy (High Availability)

SBC fault tolerant redundancy is based on a 1:1 paired protection model. For each active service card running with the SBC, there should be another service card providing failure protection (that is, standby). The same services must be provisioned on both cards (one as the primary card, one as the standby card); in this instance, the service cards are described as “paired”.

From a Cisco IOX XR system perspective, service cards are always running in active mode. SBC services running on these cards, however, run as either a primary service or standby service.

Given that SBC functionality is split among two logical service entities—the signaling border element (SBE) service and data border element (DBE) service—these services run on Cisco IOX XR service cards as follows:

- In the Unified model, SBE and DBE run on the same service card. In this case, SBE and DBE services are implemented as a single Cisco IOS XR process.
- In the Distributed model, SBE and DBE services run as separate Cisco IOX XR processes (and there may be one or more distributed DBE(s) per SBE). When running in this mode:
  - SBE and DBE services may be provisioned on different cards within the same physical device to distribute the processing load across available service cards.
  - SBE and DBE may be located on different physical devices.

Where there is no standby service card available, a software failure results in a restart of the primary SBC process. If this succeeds, the SBC process loses all call states, but management and configuration information stored in SysDb is recovered and re-applied on restart.

When there is a standby SBC provisioned, the active SBC replicates the state to the standby to provide hot standby support. The SBC process is fate shared with the Media Packet forwarder component; if one component restarts, the other component will restart.

**Note** For a complete description of commands used in this chapter, refer to the *Cisco IOS XR Session Border Controller Command Reference*. To locate documentation for other commands that appear in this chapter, use the command reference master index, or search online.

**Feature History for Implementing SBC Redundancy**

| Release | Modification |
|---|---|
| Release 3.3.0 | This feature was introduced on the Cisco XR 12000 Series Router. |

# Contents


- Prerequisites for Implementing Redundancy, page SBC-112
- Information About Implementing Redundancy, page SBC-112
- How to Implement Redundancy, page SBC-112
- Configuration Examples of Implementing a Redundancy, page SBC-116
- Additional References, page SBC-117
- Related Command Summary, page SBC-118


# Prerequisites for Implementing Redundancy

The following prerequisites are required to implement SBC redundancy:

- You must be in a user group associated with a task group that includes the proper task IDs for SBC commands being used. For detailed information about user groups and task IDs, see the defined task ID required per command in the *Cisco IOS XR Session Border Controller Command Reference*.
- You must install and activate the package installation envelope (PIE) for the SBC software.

  For detailed information about PIE installation, refer to the *Upgrading and Managing Cisco IOS XR Software* module in the *Cisco IOS XR Getting Started Guide*.
- Before implementing interworking SBC redundancy, the SBC must already be created. See the procedures described in the "SBC Configuration Prerequisites" module.

# Information About Implementing Redundancy

SBC fault tolerance is based on a 1:1 paired-protection model. For each service card running active SBC components, there can be one service card providing failure protection. The same services must be provisioned on both cards (one as the primary card, one as the standby card), and the service cards are then said to be paired. Although from an Cisco IOX XR system perspective, service cards are always running in active mode, SBC services running on these cards run as either the primary service or the standby service.

# How to Implement Redundancy

Redundancy configurations are described in the following sections:

- Configuring a Redundancy
- Deleting a Redundancy

# Configuring a Redundancy

This task configures a redundancy.

## SUMMARY STEPS

1. **configure**
2. **sbc** *service-name*
3. **service-location preferred-active** *node-id* **preferred-standby** *node-id*
4. **commit**
5. **exit**
6. **interface sbc***number*
7. **service-location preferred-active** *node-id* [**preferred-standby** *node-id]*
8. **commit**
9. **end**
10. **show services redundancy**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure**<br><br>**Example:**<br>`RP/0/0/CPU0:router# configure` | Enables global configuration mode. |
| **Step 2** | **sbc** *service-name*<br><br>**Example:**<br>`RP/0/0/CPU0:router(config)# sbc mysbc`<br>`RP/0/0/CPU0:router(config-sbc)#` | Enters the mode of an SBC service.<br>• Use the *service-name* argument to define the name of the service. |
| **Step 3** | **service-location preferred-active** *node-id* [**preferred standby** *node-id*]<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)#`<br>`preferred-active 0/2/CPU0 preferred-standby`<br>`0/4/CPU0` | Enables a service card to run SBC function as a primary, and optionally, secondary location. |
| **Step 4** | **commit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)# commit` | Saves configuration changes. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |
| **Step 5** | **exit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)# exit`<br>`RP/0/0/CPU0:router(config)#` | Exits **sbc** mode to **config** mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 6** | **interface sbc***number*<br><br>**Example:**<br>`RP/0/0/CPU0:router(config)# interface sbc1`<br>`RP/0/0/CPU0:router(config-if)#` | Enters the mode of an SBC interface, creating it if necessary. The *number* argument must be a value between 1 and 2000. |
| **Step 7** | **service-location preferred-active** *node-id* [**preferred-standby** *node-id*]<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# preferred-active 0/2/CPU0 preferred-standby 0/4/CPU0` | Enables a service card to run SBC function as a primary and, optionally, secondary location. |
| **Step 8** | **commit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# commit` | Saves configuration changes. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |
| **Step 9** | **end**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# end`<br>`RP/0/0/CPU0:router#` | Exits the configuration session. |
| **Step 10** | **show services redundancy**<br><br>**Example:**<br>`RP/0/0/CPU0:router# show services redundancy` | Shows the configured redundancy. |

# Deleting a Redundancy

This task deletes a redundancy.

## SUMMARY STEPS

1. **configure**
2. **sbc** *service-name*
3. **service-location preferred-active** *node-id* [**preferred-standby** *node-id*]
4. **commit**
5. **exit**
6. **interface sbc***number*
7. **service-location preferred-active** *node-id* [**preferred-standby** *node-id*]
8. **commit**
9. **end**
10. **show services redundancy**

## DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure**<br><br>**Example:**<br>`RP/0/0/CPU0:router# configure` | Enables global configuration mode. |
| **Step 2** | **sbc** *service-name*<br><br>**Example:**<br>`RP/0/0/CPU0:router(config)# sbc mysbc`<br>`RP/0/0/CPU0:router(config-sbc)#` | Enters the mode of an SBC service.<br>• Use the *service-name* argument to define the name of the service. |
| **Step 3** | **service-location preferred-active** *node-id* [**preferred-standby** *node-id*]<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)#`<br>`preferred-active 0/2/CPU0` | Enables a service card to run SBC function as a primary and, optionally, secondary location. |
| **Step 4** | **commit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)# commit` | Saves configuration changes. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |
| **Step 5** | **exit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-sbc)# exit`<br>`RP/0/0/CPU0:router(config)#` | Exits **sbc** mode to **config** mode. |
| **Step 6** | **interface sbc***number*<br><br>**Example:**<br>`RP/0/0/CPU0:router(config)# interface sbc1`<br>`RP/0/0/CPU0:router(config-if)#` | Enters the mode of an SBC interface, creating it if necessary. The *number* argument must be a value between 1 and 2000. |
| **Step 7** | **service-location preferred active** *node-id* [**preferred standby** *node-id*]<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# preferred-active`<br>`0/2/CPU0` | Enables a service card to run SBC function as a primary and, optionally, secondary location. |
| **Step 8** | **commit**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# commit` | Saves configuration changes. Use the **commit** command to save the configuration changes to the running configuration file and remain within the configuration session. |

| | Command or Action | Purpose |
|---|---|---|
| Step 9 | **end**<br><br>**Example:**<br>`RP/0/0/CPU0:router(config-if)# end`<br>`RP/0/0/CPU0:router#` | Exits the configuration session. |
| Step 10 | **show services redundancy**<br><br>**Example:**<br>`RP/0/0/CPU0:router#`<br>`show services redundancy` | Shows the deleted redundancy. |

# Configuration Examples of Implementing a Redundancy

This section provides the following configuration examples:

- Configuring an SBC Redundancy: Example
- Deleting the SBC Redundancy: Example

## Configuring an SBC Redundancy: Example

The following example describes a scenario in which redundant PSC-1 SBC service cards are physically located with dual route processors (RPs) and one line card in a Cisco XR 12000 Series Router. Redundant PSC-1 cards can be put in adjacent slots or non-adjacent slots.

1. Configure the SBC service redundancy:

```
config
sbc umsbc-node5
service-location preferred-active 0/2/CPU0 preferred-standby 0/4/CPU0
commit
exit
```

2. Configure SBC virtual interface redundancy:

```
interface sbc1
service-location preferred-active 0/2/CPU0 preferred-standby 0/4/CPU0
commit
```

3. Verify the SBC redundancy:

```
show service redundancy
Service type     Name                    Pref. Active        Pref. Standby
------------------------------------------------------------------------------
SBC              umsbc-node5             0/2/CPU0 Active    0/4/CPU0 Standby
```

## Deleting the SBC Redundancy: Example

The following example shows how to delete the SBC configuration created in the "Configuring an SBC Redundancy: Example" section:

1. Delete the SBC redundancy:

```
config
sbc umsbc-node5
service-location preferred-active 0/2/CPU0
commit
exit
```

2. Delete the SBC virtual interface redundancy:

```
config
interface sbc1
service-location preferred-active 0/2/CPU0
commit
```

3. Verify the SBC redundancy deletion:

```
RP/0/1/CPU0:node5#show service redundancy
Service type     Name                    Pref. Active        Pref. Standby
------------------------------------------------------------------------------
SBC              umsbc-node5             0/2/CPU0 Active
```

# Additional References

The following sections provide references related to implementing SBC redundancy (high availability).

## Related Documents

| Related Topic | Document Title |
|---|---|
| Cisco IOS XR master command reference | *Cisco IOS XR Master Commands List* |
| Cisco IOS XR SBC interface configuration commands | *Cisco IOS XR Session Border Controller Command Reference* |
| Initial system bootup and configuration information for a router using the Cisco IOS XR Software | *Cisco IOS XR Getting Started Guide* |
| Cisco IOS XR command modes | *Cisco IOS XR Command Mode Reference* |

## Standards

| Standards | Title |
|---|---|
| No new or modified standards are supported by this feature, and support from existing standards has not been modified by this feature. | — |

## MIBs

| MIBs | MIBs Link |
|---|---|
| The SBC MIB is not supported for Release 3.3.0. | To locate and download MIBs for selected platforms using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL:<br>http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml |

## RFCs

| RFCs | Title |
|---|---|
| No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. | — |

## Technical Assistance

| Description | Link |
|---|---|
| The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content. | http://www.cisco.com/techsupport |

# Related Command Summary

This section provides an alphabetical list of the commands related to redundancy configuration on the Cisco XR 12000 Series Router. For more information about the commands, see the *Cisco IOS XR Session Border Controller Command Reference.*

| Command | Purpose |
|---|---|
| **service-location preferred-active** *node-id* **preferred-standby** *node-id* | Enables a service card to run SBC function as a primary, and optionally, secondary location. |
| **interface sbc***number* | Enters the mode of an SBC interface, creating it if necessary. The *number* argument must be a value between 1 and 2000. |
| **show services redundancy** | Shows the redundancies running on the system. |